DTU Library

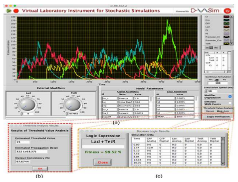
D-VASim: A Software Tool to Simulate and Analyze Genetic Logic Circuits

Baig, Hasan; Madsen, Jan

Publication date:
2016

Document Version
Publisher's PDF, also known as Version of record

D-VASim: A Software Tool to Simulate and Analyze Genetic Logic Circuits



"Using LabVIEW, we reduced our work tremendously to make D-VASim an attractive and efficient tool for the simulation and analysis of genetic logic circuits. This project can help students, biologists, biotechnologists, and genetic circuit design engineers to perform wet lab experiments virtually, but it may also initialize a new research problem of timing analysis in the domain of genetic design automation."

- Hasan Baig, Technical University of Denmark (http://www.dtu.dk/english)

Read the Full Case Study

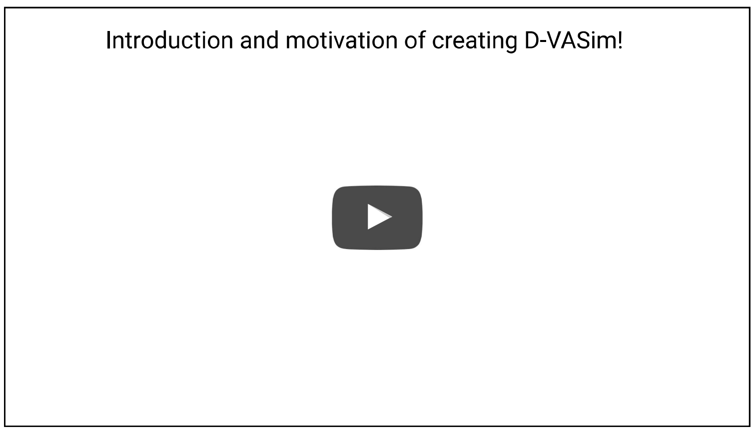
The Challenge:

Creating a software tool for the simulation and analysis of genetic logic circuits to help researchers performing wet lab experiments virtually, because the manual process of wet lab experimentation of genetic logic circuits is time consuming and a challenging task for early-stage researchers with limited experience in the field of biology.

The Solution:

Using LabVIEW to develop a user-friendly simulation tool named Dynamic Virtual Analyzer and Simulator (D-VASim), which is the first software tool in the domain of synthetic biology that provides a virtual laboratory environment to perform run-time interactive simulation and analysis of genetic logic circuits.

Author(s):
Hasan Baig - Technical University of Denmark (http://www.dtu.dk/english)
Jan Madsen - Technical University of Denmark (http://www.dtu.dk/english)



Background

Technical University of Denmark (DTU) has been dedicated to fulfilling the vision of H.C. Ørsted, the father of electromagnetism and founder of DTU, which was to benefit society through the development of natural and technical sciences. Today, DTU continues to set new records and develop partnerships with industry, helping DTU rank at the forefront of technical universities in Europe.

The Embedded Systems Engineering (ESE) group is a research section within DTU Compute, Denmark's largest environment for Mathematics and Computer Science. We conduct research on a broad range of topics central to the design of modern embedded systems, including real-time systems, fault-tolerant and safety-critical systems, and concurrent and parallel programming. In addition, we research a range of models, methods, and tools for the analysis, design, and verification of such systems. We have also been working on state-of-the-art research topics including microfluidic biochips and genetic design automation (GDA) for the applications of synthetic biology.

Genetic logic circuits are an application of synthetic biology in which parts of the DNA inside a living cell are re-engineered to perform certain Boolean functions, mimicking those observed in digital electronics. Unlike digital electronic circuits, which are made up of digital logic gates, genetic logic circuits are composed of biological components of DNA. These circuits have a wide range of applications in biomedical and biotech industries, including gene regulation systems, drug development, cellular computation, stem cell reprogramming, gene therapy, and more. By implementing rational, controllable logic elements in cellular systems, researchers can use living systems as engineered "machines" to perform a vast range of useful functions. For example, the most common treatment of cancer is radiation therapy, which uses high-energy particles to destroy cancer cells. One of the side effects of this treatment is that it not only destroys cancerous cells, but the nearby normal cells as well, causing patients to lose their hair. The development of a genetic sensor circuit to identify the defective cancerous cells and kill them on the spot, without killing neighboring normal cells, is an interesting application of synthetic biology.

Engineers and scientists develop these circuits in wet labs through repeated in-vitro experiments, which is a tedious and time consuming process. Therefore, they usually develop the computational models of these genetic circuits prior to wet lab experiments to perform the analysis in-silico (in computer).

The Systems Biology Markup Language (SBML) is a standard way of representing biological models in a computerized form. Engineers and scientists can use the machine-readable format to share and publish models in a form that can be used by different software tools. Users can take advantage of the developed GDA tools for the model construction and analysis. A vast majority of these tools support reading and/or writing SBML files. Some of these tools serve as a toolbox for commercial platforms including The MathWorks, Inc. MATLAB® software , Mathematica, and Oracle. Others are developed as APIs or plugins to specific software systems, while the rest are independent tools for design and simulation including CellDesigner, iBioSim, and COPASI. None of them, however, are capable of letting the user interact with the model during the running of the simulation. No publically available tool exists that allows users to interact with the model on the fly during the simulation, effectively creating a virtual laboratory environment.

Dynamic Virtual Analyzer and Simulator (D-VASim)

As mentioned above, wet lab experimentation is a time consuming process, and due to mishandling of the apparatus or other errors, scientists must often repeat the experiments. To avoid lengthy wet lab experiments, these biological models are commonly validated through in-silico analysis either by solving reaction kinetics using ordinary differential equations or by performing more realistic analysis through stochastic simulations. D-VASim is the first simulation tool that empowers the user to perform experiments in simulation by varying different parameters and observing their effects on the circuit model during run time, instead of repeating time consuming experiments in a wet lab to find out the desired set of parameters. It is also a first software tool, which supports the preliminary timing analysis of genetic logic circuits. Figure 1 depicts the complete simulation and analysis flow of D-VASim.

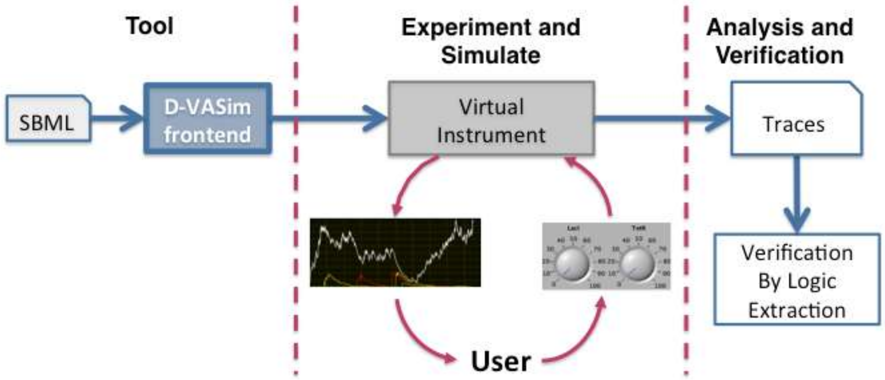


Figure 1. Virtual Simulation and Analysis Flow of D-VASim

D-VASim takes an SBML model of genetic logic circuit as an input and lets the user analyse the components of the model in a user-friendly manner, as shown in Figure 2.

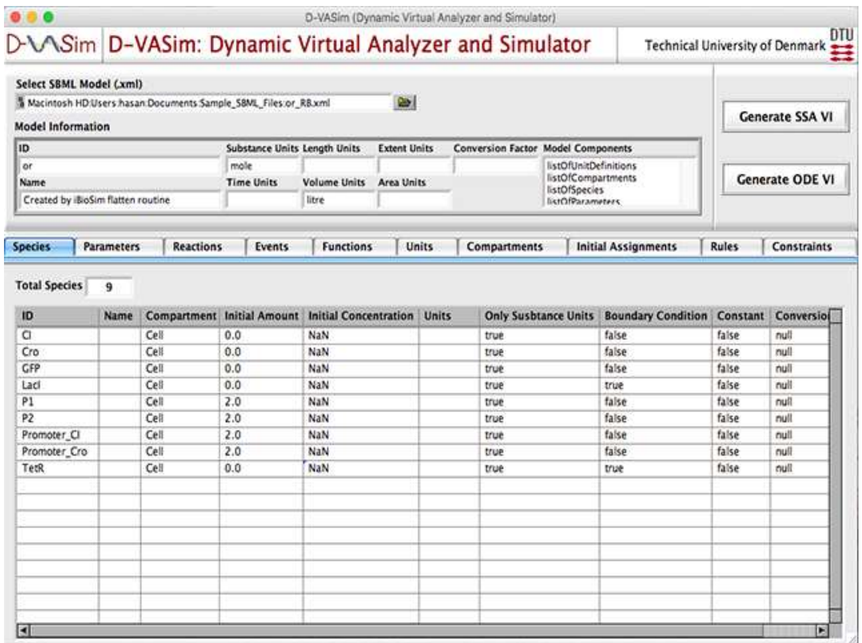


Figure 2. D-VASim Front End User Interface

For each logic circuit model, D-VASim generates a separate virtual laboratory environment to perform the deterministic and the stochastic simulation. Figure 3(a) shows a virtual environment for stochastic simulations generated by pressing a button Generate SSA VI, shown in Figure 2. This virtual environment serves as a standalone virtual instrument or virtual apparatus for the specific logic circuit model, so users can interact with the model, observe its behaviour, and make the direct changes in the concentration of input species, all during run time. This gives users a feeling of performing live experiments in a wet lab by adding molar concentration of input species and observing their effects on the output species. Besides providing users the ability to interact with the model during run time, D-VASim can also analyze the threshold values and propagation delays of genetic logic circuits (Figure 3(b)). Moreover, we can use D-VASim to verify the Boolean logic function of a genetic logic circuit model (Figure 3(c)). D-VASim also allows users to analyse the timings of genetic logic circuits using digital waveforms, shown in Figure 4.

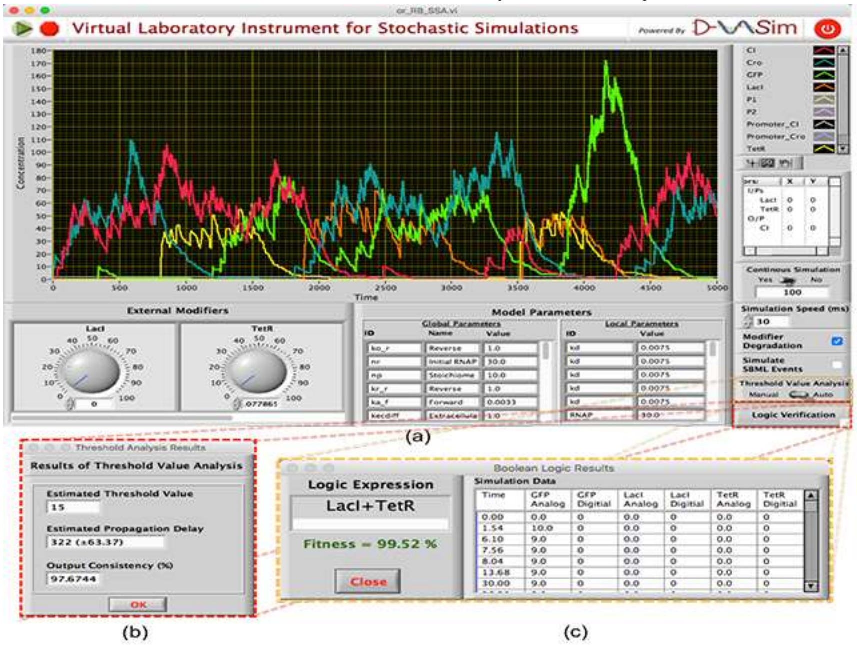


Figure 3. Stochastic Simulation Screenshots of D-VASim (a) VI Generated by D-VASim to Perform Stochastic Simulations of a Genetic Circuit (b) Threshold and Propagation Analysis Results Automatically Estimated by D-VASim (c) Logic Verification and Analysis Done by D-VASim

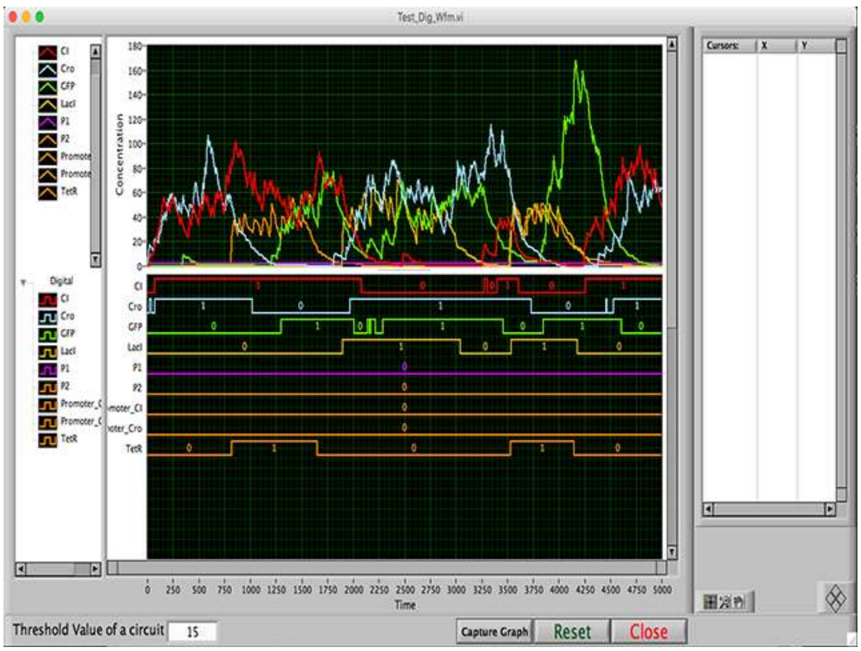
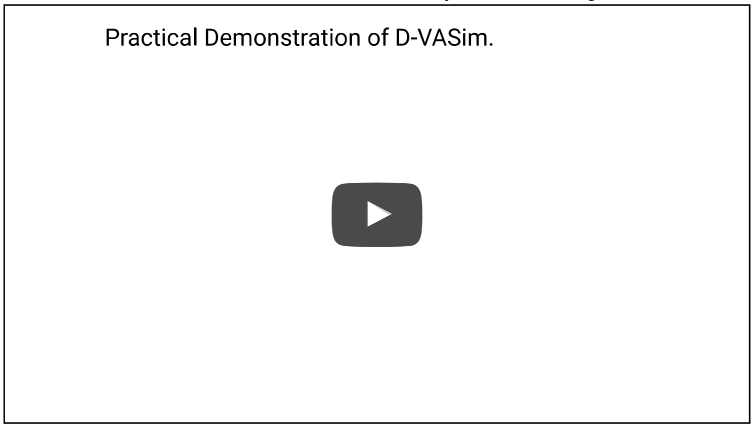


Figure 4. Mixed-Signal Waveforms for Analyzing the Timings of a Genetic Logic Circuit

We used LabVIEW (http://www.ni.com/labview/) software to develop D-VASim. We chose LabVIEW to develop this simulation platform because of its intuitive graphical programming environment, rapid application development, easy debugging, and the tremendously reduced efforts required for graphical user interface (GUI) development. We also found built-in function libraries, file I/O functions, and the possibility of hybrid programming vital to the success of this project. The customizable GUI was imperative towards an interactive, intuitive simulation environment.

The rapid application development in LabVIEW helped us to release D-VASim for public use in a very short amount of time. If we had chosen any other programming platform, it may have taken another six months or more to get to the point where D-VASim is currently. LabVIEW further benefited us in making D-VASim available for multiple operating systems (Windows and Mac) using the same source code. Furthermore, the self-paced online LabVIEW core courses from NI helped elevate us in the process of learning and development.



Conclusion

Using LabVIEW, we reduced our work tremendously to make D-VASim an attractive and efficient tool for the simulation and analysis of genetic logic circuits. This project can help students, biologists, biotechnologists, and genetic circuit design engineers to perform wet lab experiments virtually, but it may also initialize a new research problem of timing analysis in the domain of genetic design automation.

High-tech industry researchers and university students can use D-VASim to simulate the genetic models prior to wet lab work. Users can perform virtual wet lab experiments, as well as timing and logic analysis of genetic circuits. We work continuously with Massachusetts Institute of Technology and Boston University to characterize and validate the parameters of genetic logic components and combinatorial genetic circuits using D-VASim. This speeds up the process of finding the correct set of a genetic model's parameter required to obtain the desired circuit behavior.

Author Information:

Hasan Baig

Technical University of Denmark (http://www.dtu.dk/english)

Richard Petersens Plads, Building 322, Room 113, Technical University of Denmark 2800

Kongens Lyngby

Danmark

Tel: +45 45 25 30 33

haba@dtu.dk (mailto:haba@dtu.dk)